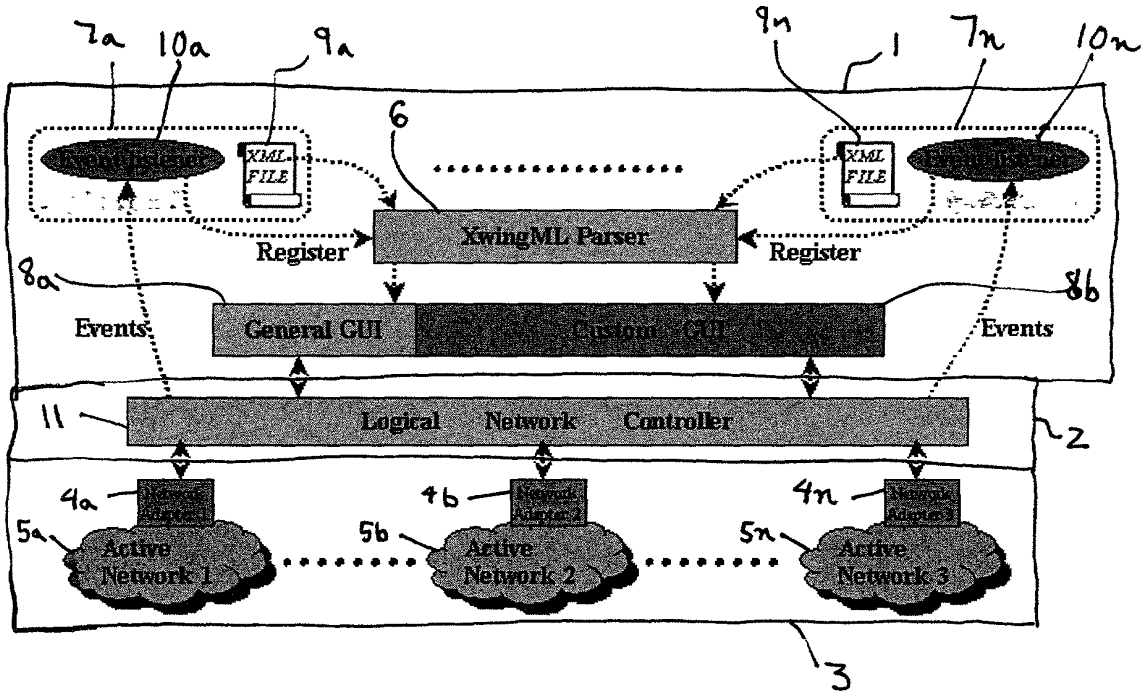
7a
10a
9a
9n
1
7n
10n
6
XML FILE
XML FILE
XwingML Parser
Register
Register
8a
8b
Events
General GUI
Custom GUI
Events
11
Logical Network Controller
2
4a
4b
4n
5a
5b
5n
Active Network 1
Active Network 2
Active Network 3
3

FIGURE 1

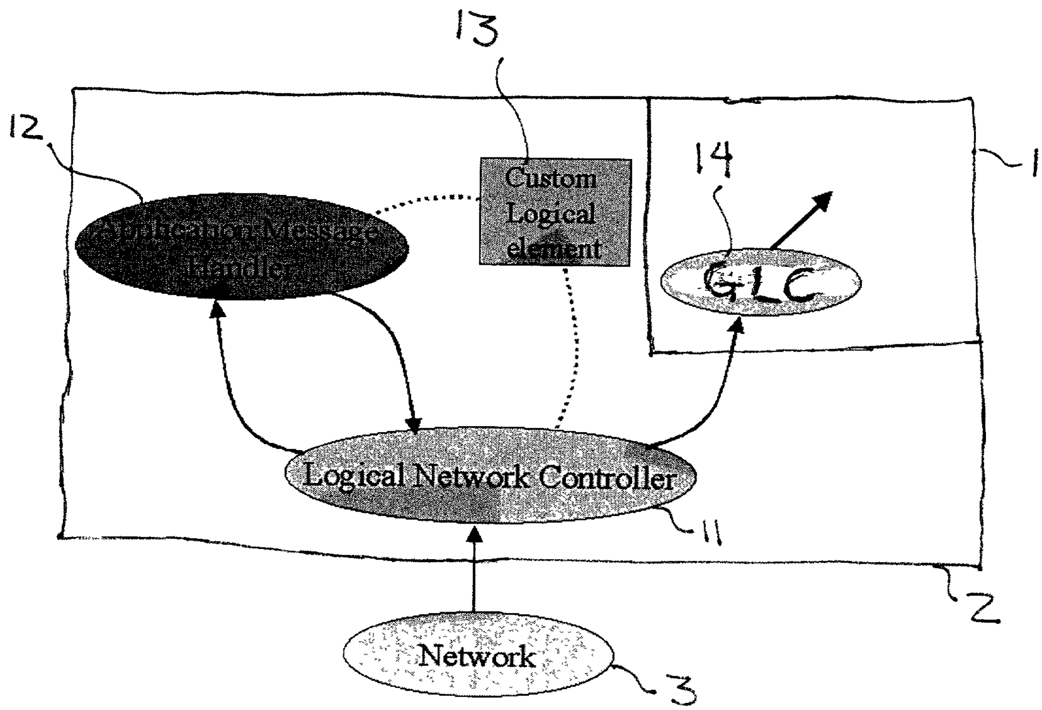
13
12
Custom
Logical
element
14
1
Application Message
Handler
GLC
Logical Network Controller
11
2
Network
3

FIGURE 2

```
<?xml version="1.0"?>
<!DOCTYPE XwingML SYSTEM "file:///usr/XwingML/xml/xwingml.dtd">
<XwingML>
   <Classes>
     <Instance name="listener" className="answer.GuiListener"/>
     <!-- a single listener for all messages. -->
     <MessageHandler  className="answer.AnswerMessageHandler" />
   </Classes>

   <TopoSpec  name="region to display topology" height="60%"
width="70% title="Topology region">
    <NodeSpec type="client" objectClassName="answer.TopologyHost" />
    <NodeSpec type="router" objectClassName="answer.TopologyRouter" />
    <NodeSpec type="server" objectClassName="answer.TopologyServer" />
    <LinkSpec type="*" color="black" width="5"/>
    <LinkSpec type="RoutingTreeEdge" color="red"/>
   </TopoSpec>

   <JFrame name="MainFrame" title="Demo"  x="10%" y="10%"
           width="40%" height="42%" background="white" >
     <BoxLayout axis="Y_AXIS"/>

     <JPanel name="panel1" gridx="0" gridy="0" fill="VERTICAL" weighty="25">
        <JLabel name="textLabel" text="Generic Active Network Monitoring Tool"
                fontSize="15" foreground="blue"/>
     </JPanel>
     <TopoRegion/>
     <JPanel name="panel2" gridx="0" gridy="2" fill="VERTICAL" weighty="25">
        <JButton name="cancel" text="QUIT" actionListener="listener"/>
     </JPanel>

   </JFrame>
</XwingML>
```

FIGURE 3

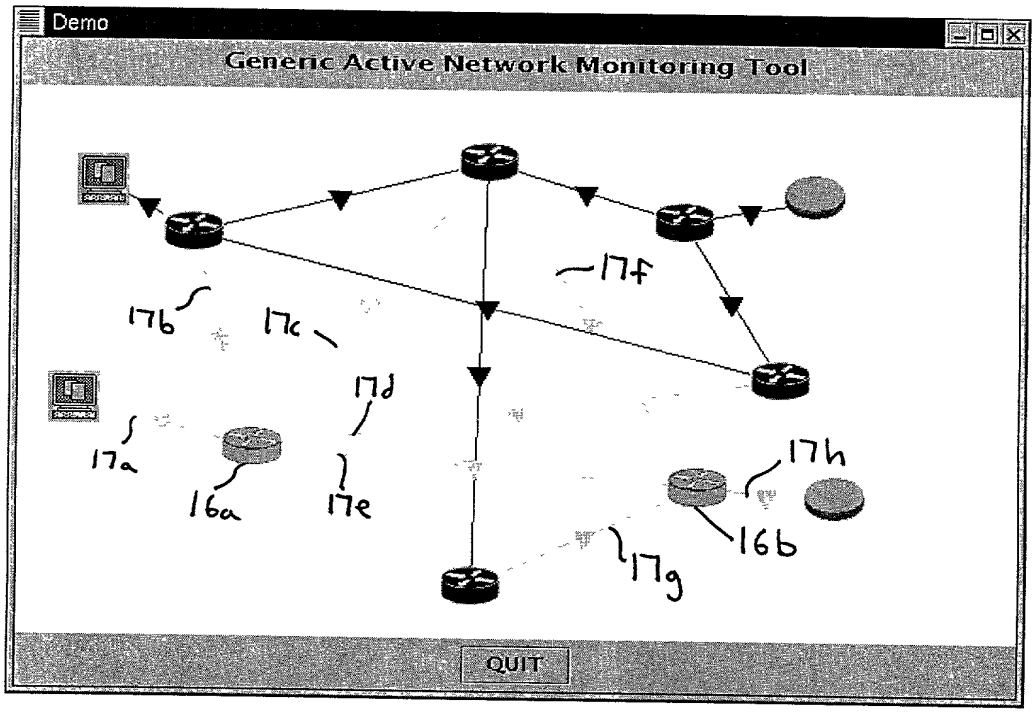
Demo
Generic Active Network Monitoring Tool
17b
17c
17f
17d
17a
16a
17e
17h
16b
17g
QUIT

FIGURE 4

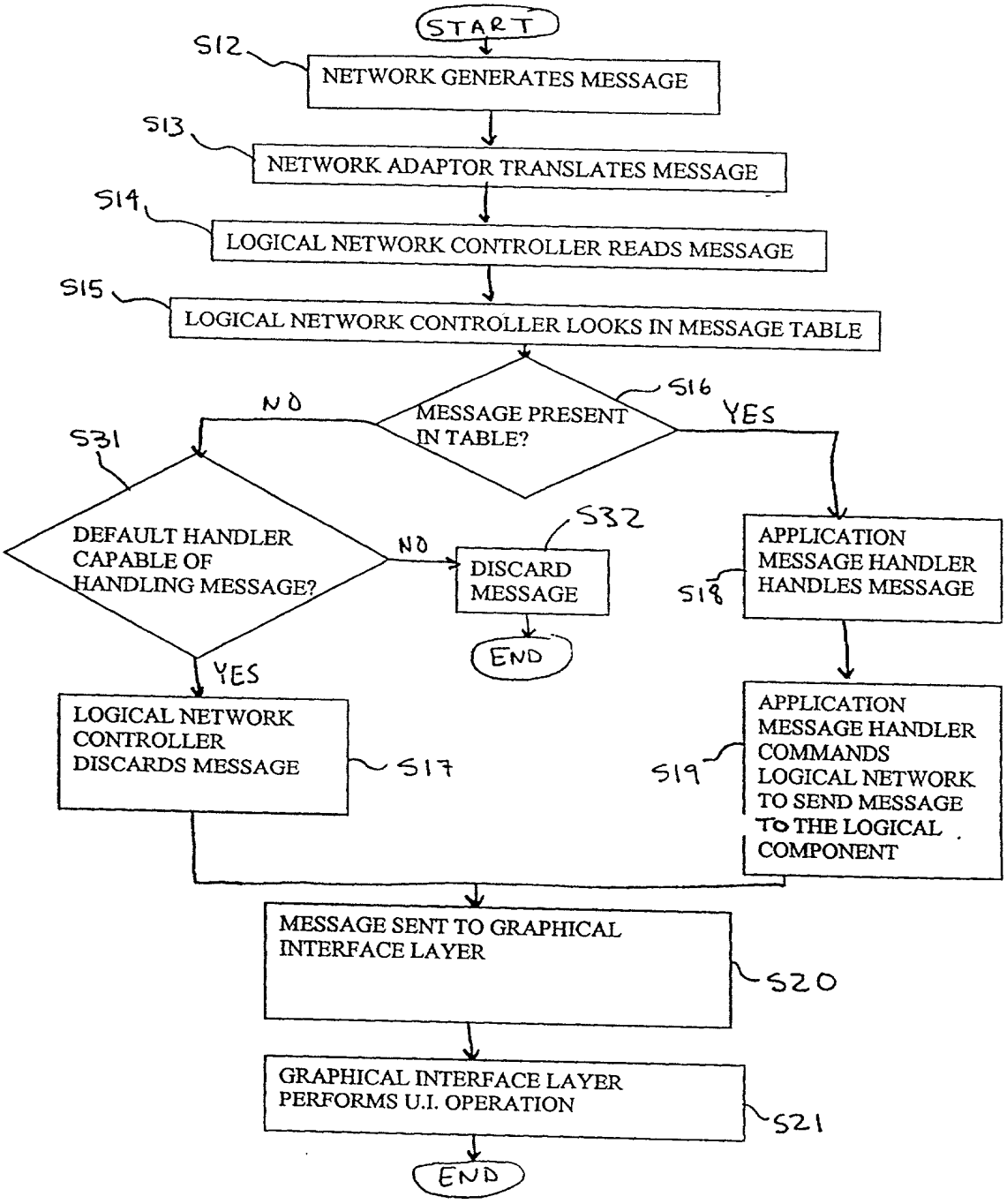
START
S12
NETWORK GENERATES MESSAGE
S13
NETWORK ADAPTOR TRANSLATES MESSAGE
S14
LOGICAL NETWORK CONTROLLER READS MESSAGE
S15
LOGICAL NETWORK CONTROLLER LOOKS IN MESSAGE TABLE
S16
MESSAGE PRESENT IN TABLE?
NO
YES
S31
DEFAULT HANDLER CAPABLE OF HANDLING MESSAGE?
NO
S32
DISCARD MESSAGE
END
YES
LOGICAL NETWORK CONTROLLER DISCARDS MESSAGE
S17
S18
APPLICATION MESSAGE HANDLER HANDLES MESSAGE
S19
APPLICATION MESSAGE HANDLER COMMANDS LOGICAL NETWORK TO SEND MESSAGE TO THE LOGICAL COMPONENT
MESSAGE SENT TO GRAPHICAL INTERFACE LAYER
S20
GRAPHICAL INTERFACE LAYER PERFORMS U.I. OPERATION
S21
END

FIGURE 5(a)

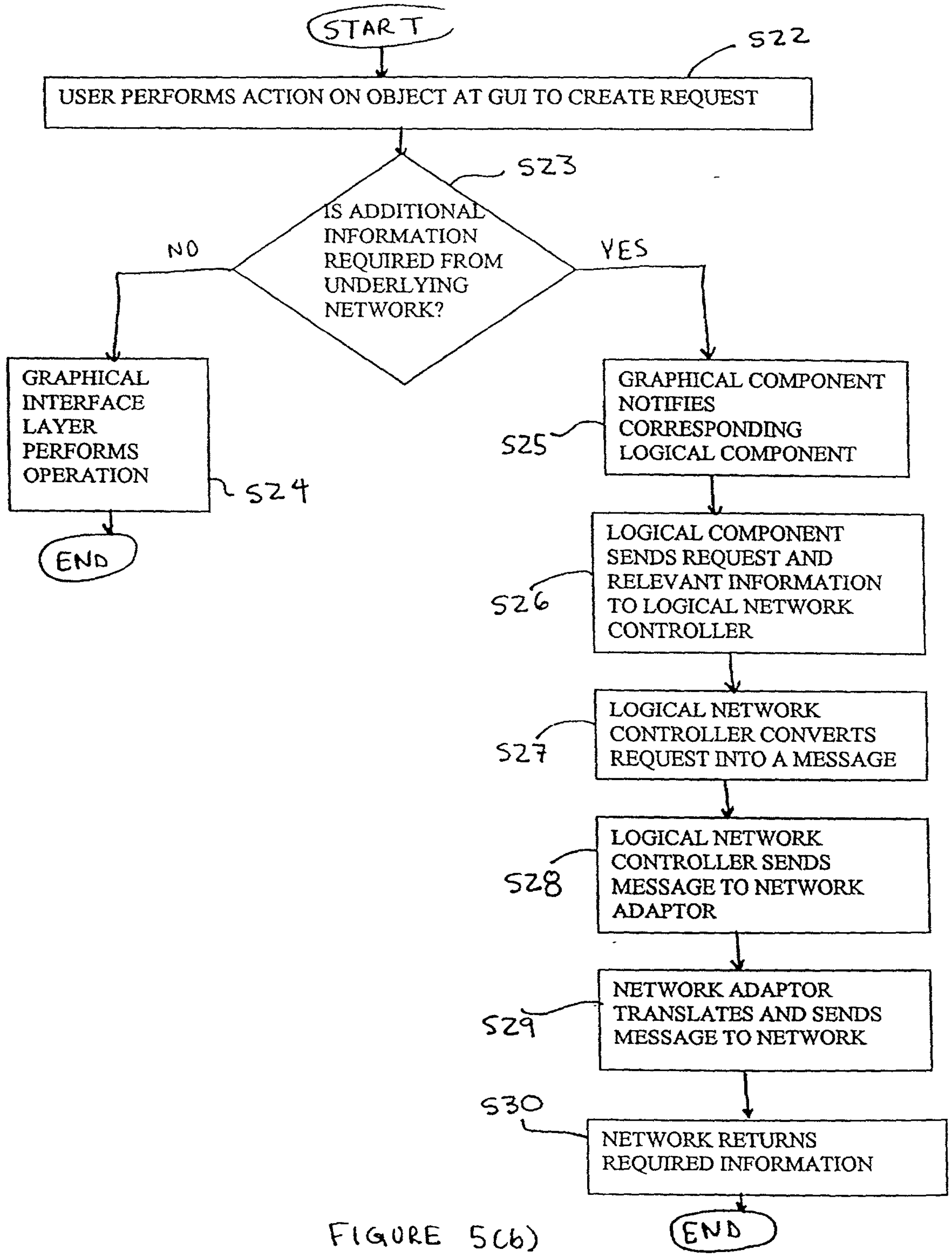

FIGURE 5(b)

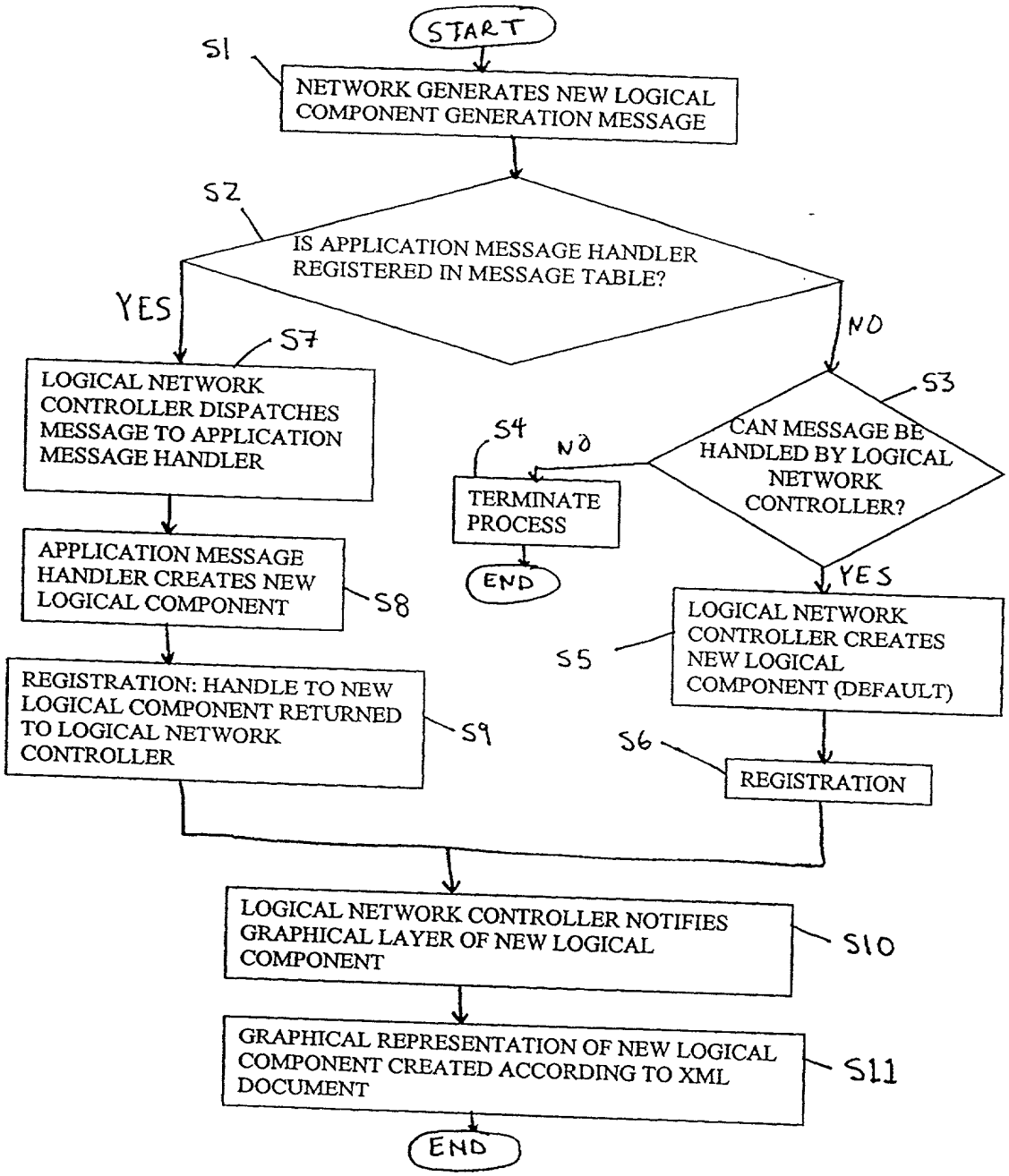
START
S1
NETWORK GENERATES NEW LOGICAL COMPONENT GENERATION MESSAGE
S2
IS APPLICATION MESSAGE HANDLER REGISTERED IN MESSAGE TABLE?
YES
NO
S7
LOGICAL NETWORK CONTROLLER DISPATCHES MESSAGE TO APPLICATION MESSAGE HANDLER
APPLICATION MESSAGE HANDLER CREATES NEW LOGICAL COMPONENT
S8
REGISTRATION: HANDLE TO NEW LOGICAL COMPONENT RETURNED TO LOGICAL NETWORK CONTROLLER
S9
S3
CAN MESSAGE BE HANDLED BY LOGICAL NETWORK CONTROLLER?
S4
NO
TERMINATE PROCESS
END
YES
S5
LOGICAL NETWORK CONTROLLER CREATES NEW LOGICAL COMPONENT (DEFAULT)
S6
REGISTRATION
LOGICAL NETWORK CONTROLLER NOTIFIES GRAPHICAL LAYER OF NEW LOGICAL COMPONENT
S10
GRAPHICAL REPRESENTATION OF NEW LOGICAL COMPONENT CREATED ACCORDING TO XML DOCUMENT
S11
END

FIGURE 6